



Inteligencia Artificial para Videojuegos

Grado en Desarrollo de Videojuegos

Proyecto final

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid



Proyecto: **Animal Shelter**

Fecha del enunciado: 25 de mayo de 2021

Fecha de defensa: 15 de junio de 2021

Fecha de entrega: 16 de junio de 2021

Importante: Haz la entrega en tiempo y forma, subiendo al campus virtual un fichero *IAVFinal-Apellidos.txt* sobre el proyecto final. Dentro de él incluye los datos y contribuciones de los alumnos implicados, una breve explicación del proyecto, los problemas existentes y el enlace al repositorio donde están: el fichero *README.md* con el índice de la documentación técnica, la carpeta *IAVFinal-Apellidos* con el proyecto (plugins, recursos y el código fuente), la versión ejecutable para Windows de 64bits *IAVFinal-Apellidos.exe* (con sus carpetas y ficheros acompañantes) y el video comentado con las pruebas *IAVFinal-Apellidos.mp4*.

1. Introducción

Motivación:

“El estudio del comportamiento de los animales de distintas especies y la convivencia entre ellos resulta algo digno de observar. Sus acciones están normalmente motivadas por instintos y sus capacidades, además de resultar curiosas, podrían sorprender al humano.”

Contexto:

El jugador es un joven investigador que se aventura a rescatar animales en una serie de islas remotas y llevarlos a un refugio en una isla segura. Su motivación principal es estudiar y observar a los animales que rescata y aprender cómo estos pueden llegar a convivir o qué especies no pueden ni mantenerse cerca.

Desarrollo de proyecto:

Para este proyecto el jugador se encuentra en un refugio de animales situado en una isla, pues es el encargado de buscar y rescatar animales salvajes de las islas vecinas. Se deberá desarrollar la IA de los distintos animales, sus comportamientos, sus decisiones y sus acciones. Estos también podrán interactuar con el jugador.

2. Planteamiento del proyecto

Desarrollar la IA de los animales de distintas especies con ayuda de la herramienta Bolt y si es necesario, Scripting en C#. Los animales deben ser capaces de tomar decisiones en base a sus instintos o motivaciones. El jugador deberá domesticar a los animales salvajes en las islas remotas ganándose su confianza con comida, solo entonces podrá llevárselos al refugio. Una vez en el

refugio los animales merodearán por la isla segura y dependiendo de su cerebro inteligente, realizarán ciertas acciones.

Se utiliza una malla de navegación para el movimiento de los animales y gracias a está, los obstáculos (árboles en su mayoría) de la isla pueden ser sorteados por los animales. Para evitar situaciones muy complejas:

- Los animales deben evitar **peleas** con animales que ya están enfrentándose a una.
- Solo se puede **rescatar** a un animal cada vez que se sale a explorar una nueva isla.
- Si se añaden **obstáculos** se deben incluir en la malla de navegación para facilitar el movimiento a los animales.
- Los animales cuentan con un **nivel** de **peligrosidad** y dependiendo de este atributo, considerarán si enfrentarse a ciertos animales.

La entrega será realizada en tiempo y forma **[1 pto.]**, el proyecto estará bien diseñado, organizado y comentado **[1 pto.]**, y la documentación explicará con claridad cuáles fueron las técnicas implementadas **[1 pto.]**, las pruebas realizadas y los resultados obtenidos **[1 pto.]**.

El prototipo ejecutable será usable y funcional, permitiendo:

- Viajar con un botón para **explorar** una isla en busca de animales salvajes, y una vez encontrarlos, poder alimentarlos hasta conseguir domesticarlos **[1 pto.]**.
- Observar el progreso al **domesticar** cada animal con una barra azul y los puntos de vida también representados con una barra **[1 pto.]**.
- Ver como los animales pueden tener algo en **mente** y a partir de ahí desarrollar distintas **decisiones** dentro del refugio, para facilitar esto se debe añadir **feedback** visual como un símbolo que represente lo que se tiene en mente **[1 pto.]**.
- Llevar una especie de mochila con **alimentos** para los animales y que, al tirarlos, estos puedan identificarlos (pueden no ser comestibles para una especie de animal, en ese caso el animal los ignora) **[1 pto.]**.
- Presenciar un **enfrentamiento** entre animales dentro del refugio **[1 pto.]**, solo uno sale con vida y dependiendo de la **motivación** que ha llevado al enfrentamiento (por simple **hostilidad** o por **hambre** de uno de los animales), el animal victorioso **devorará** al vencido o simplemente dejará su cadáver **[1 pto.]**.

3. Restricciones y consejos

A la hora de desarrollar este proyecto es obligatorio:

- Utilizar únicamente las herramientas de Unity y opcionalmente los plugins de terceros *Bolt* o *Behavior Designer*, sin reutilizar código ajeno al que proporciona el profesor.
- Documentar claramente los algoritmos, heurísticas o cualquier “truco” utilizado.

- Diseñar y programar de la manera más limpia y elegante posible, separando la parte visual e interactiva del juego, del modelo y las técnicas de IA implementados.
- Evitar, en la medida de lo posible, el uso de recursos audiovisuales pesados o ajenos.

Pensando tanto en las pruebas como en la revisión del profesor, y también con ánimo de reutilizar el esfuerzo de desarrollo, conviene crear herramientas visuales cómodas para mostrar escenarios de ejemplo interesantes y con instrucciones de uso, etc. El manejo debe ser ágil e intuitivo para poder repetir rápidamente todas las pruebas que sean necesarias con las variaciones que hagan falta.

4. Referencias y ampliaciones

Como punto de partida para la investigación, además de la bibliografía de la asignatura, puedes utilizar las siguientes referencias. En ningún caso debes replicar el código que encuentres por ahí; asegúrate de entenderlo y verifica que funciona *exactamente* como pide este enunciado.

- Bolt, Visual Scripting
- Opsive, Behavior Designer
- Unity, Navegación y Búsqueda de caminos
- Unity 2018 Artificial Intelligence Cookbook, Second Edition (Repositorio)
- Unity Artificial Intelligence Programming, Fourth Edition (Repositorio)